

Monadic simultaneous rigid E -unification

Yuri Gurevich^{1,a}, Andrei Voronkov^{*,b}

^a EECS Department, University of Michigan, Ann Arbor, MI 48109-2122, USA

^b Computing Science Department, Uppsala University, Box 311, S-751 05 Uppsala, Sweden

Received July 1997; revised February 1998

Communicated by A.A. Letichevsky

Abstract

We study the monadic case of a decision problem known as simultaneous rigid E -unification. We show its equivalence to an extension of word equations. We prove decidability and complexity results for special cases of this problem. © 1999 Elsevier Science B.V. All rights reserved

Keywords: Simultaneous rigid E -unification; Word equations; Regular constraints

1. Introduction

Simultaneous rigid E -unification is a combinatorial problem in equational logic, which is closely connected with some formulations of the Herbrand theorem and with automated theorem proving by the tableau method and the connection (or mating) method [16, 17]. In this section we define simultaneous rigid E -unification, discuss its connection with several decision problems in logic, and survey some known results.

We shall consider *equational logic*, i.e. logic whose only predicate is the equality predicate \simeq , but our results can easily be extended to general first-order logic with predicates other than equality. Let $s_1, t_1, \dots, s_n, t_n, s, t$ be terms. All atomic formulas in equational logic are *equations*, i.e. expressions of the form $s \simeq t$. We do not distinguish an equation $s \simeq t$ from the equation $t \simeq s$. We write $s_1 \simeq t_1, \dots, s_n \simeq t_n \vdash s \simeq t$ to denote that the formula $\forall(s_1 \simeq t_1 \wedge \dots \wedge s_n \simeq t_n \supset s \simeq t)$ is true, i.e. it is provable in first-order (classical or intuitionistic) logic. Equivalently, we can say that s and t lie in the same class of the congruence induced by $\{s_1 \simeq t_1, \dots, s_n \simeq t_n\}$.

¹ Partially supported by grants from NSF, ONR and the Faculty of Science and Technology of Uppsala University.

* Corresponding author. Supported by a TFR grant. E-mail: voronkov@csd.uu.se.

A *rigid equation* is an expression $\mathcal{E} \vdash_{\forall} s \simeq t$, where \mathcal{E} is a finite set of equations. The set \mathcal{E} is called the *left side* of this rigid equation, and the equation $s \simeq t$ – its *right side*. A *solution to a rigid equation* $\{s_1 \simeq t_1, \dots, s_n \simeq t_n\} \vdash_{\forall} s \simeq t$ is any substitution θ such that $s_1\theta \simeq t_1\theta, \dots, s_n\theta \simeq t_n\theta \vdash s\theta \simeq t\theta$. A *system of rigid equations* is a finite set of rigid equations. A *solution to a system of rigid equations* \mathcal{R} is any substitution that is a solution to every rigid equation in \mathcal{R} . The problem of solvability of rigid equations is known as *rigid E-unification*. The problem of solvability of systems of rigid equations is known as *simultaneous rigid E-unification*, in the sequel abbreviated as SREU.

We shall denote sets of equations by \mathcal{E} , systems of rigid equations by \mathcal{R} , and rigid equations by R . We shall sometimes write the left side of a rigid equation as a *sequence* of equations, for example, $x \simeq a \vdash_{\forall} g(x) \simeq x$ instead of $\{x \simeq a\} \vdash_{\forall} g(x) \simeq x$.

In [2] it is shown that the decidability of SREU is equivalent to the decidability of some other fundamental problems, for example, the following ones:

Problem 1 (Formula Instantiation). *Given a quantifier-free formula $\varphi(\bar{x})$, is there a term sequence \bar{t} such that the formula $\varphi(\bar{t})$ is provable?*

Problem 2 (Existential Intuitionistic). *Is a given existential formula $\exists \bar{x} \varphi(\bar{x})$ provable in intuitionistic logic?*

Problem 3 (Prenex Intuitionistic). *Is a given prenex formula provable in intuitionistic logic?*

For a suitable notion of a derivation skeleton, SREU is also equivalent to the following problem.

Problem 4 (Skeleton Instantiation). *Given a formula φ and a derivation skeleton, is there a derivation of φ having this skeleton?*

Some known results on SREU are the following.

- SREU is undecidable [4]. This result implies that Problems 1–4 are undecidable.
- SREU with ground left sides is undecidable [12].
- SREU with ground left sides and two variables is undecidable [15].
- SREU with one variable is DEXPTIME-complete [1].

The last two results imply a complete classification of decidable prenex fragments of intuitionistic predicate calculus with equality: the $\exists\exists$ fragment is undecidable and the $\forall^*\exists\forall^*$ fragment is decidable. All the above-mentioned undecidability results require that the signature contain a function symbol of arity ≥ 2 .

When all function symbols have arity ≤ 1 , Problems 1–4 are equivalent to *monadic SREU*, i.e. SREU in the signature where all function symbols have arity ≤ 1 .

The decidability of monadic SREU is an open problem. The following facts are known about monadic SREU [3].

- The word equation problem is effectively reducible to monadic SREU.
- Monadic SREU with one function symbol is decidable (this fact has a non-elementary proof).

- Monadic SREU is decidable if and only if it is decidable in the signature with two unary function symbols.

This paper studies monadic SREU. Although the general case remains an open problem, we prove its equivalence to a combinatorial problem on words defined in Section 5. This problem is defined in terms of *ideals* on the set of pairs of words and called *the ideal membership problem*. We prove

Theorem 4. *Monadic SREU is decidable if and only if the ideal membership problem is decidable.*

We also prove the decidability of some special cases of monadic SREU. In Section 4 we prove

Theorem 3. *Monadic SREU with one variable is PSPACE-complete.*

The result uses essentially the same technique as the main result of [1]: the decidability of SREU with one variable, but instead of tree automata used in [1] we use finite automata.

As we already mentioned, Plaisted [12] proved that SREU with ground left sides is undecidable. The corresponding monadic case is shown to be decidable in Section 3:

Theorem 2. *Monadic SREU with ground left sides is decidable.*

The complexity of monadic SREU with ground left sides is not known. We prove

Theorem 1. *Monadic SREU with one variable and ground left sides is PSPACE-complete.*

2. Preliminaries

In this section we introduce basic definitions concerning terms, equations, words, word equations, automata, and rewrite rules. We have to define so many concepts since it is unreasonable to expect a reader to know everything. We also assert some statements proved elsewhere and prove some properties of the introduced notions, which will be used in subsequent sections.

The symbol \Leftarrow means “equal by definition”.

2.1. Terms and equations

The set of all variables of a term t is denoted $var(t)$. A term is *ground* if it has no variables, i.e. $var(t) = \emptyset$. The symbol \vdash denotes provability in first-order logic. When we write $\varphi_1, \dots, \varphi_n \vdash \varphi$, where $\varphi_1, \dots, \varphi_n, \varphi$ are formulas, it means provability of the formula $\varphi_1 \wedge \dots \wedge \varphi_n \supset \varphi$. *Substitutions* of terms t_1, \dots, t_n for variables

x_1, \dots, x_n are denoted $\{t_1/x_1, \dots, t_n/x_n\}$. The *application of such a substitution θ to a term t* , is the operation of simultaneous replacement of all occurrences of x_i by t_i . The result of the application is the term denoted $t\theta$. We shall also apply substitutions to equations and sets of equations, and use the same notation for the result of the application.

The *domain* and the *range* of a substitution θ , denoted by $\text{dom}(\theta)$ and $\text{ran}(\theta)$, respectively, are defined as

$$\text{dom}(\theta) \rightleftharpoons \{x \mid x \text{ is a variable and } x\theta \neq x\},$$

$$\text{ran}(\theta) \rightleftharpoons \{x\theta \mid x \in \text{dom}(\theta)\}.$$

For every expression E (for example, term, or a set of equations), we denote by E_c^t the expressions obtained from E by the replacement of all occurrences of the constant c by a term t . We write $s[t]$ to denote a particular occurrence of a subterm t of a term s .

In this paper, we shall only consider *monadic signatures*. Any such signature consists of a finite set \mathcal{F} of unary function symbols and a finite set \mathcal{C} of constants, and will be denoted $\mathcal{F} \cup \mathcal{C}$. The set of ground terms of a signature $\mathcal{F} \cup \mathcal{C}$ is denoted by $T_{\mathcal{F} \cup \mathcal{C}}$. We always assume $\mathcal{C} \neq \emptyset$ and hence $T_{\mathcal{F} \cup \mathcal{C}} \neq \emptyset$. For every set of equations \mathcal{E} we denote by $T(\mathcal{E})$ the set of all terms occurring in \mathcal{E} and their subterms. For example, if $\mathcal{E} = \{f(x) \simeq g(c), c \simeq g(f(x))\}$, then $T(\mathcal{E}) = \{x, f(x), c, g(c), g(f(x))\}$.

We shall denote variables by x, y, z , constants by a, b, c, d , function symbols by f, g, h , terms by r, s, t , and substitutions by θ .

We shall use the following statement

Lemma 2.1. *Given a finite set of equations \mathcal{E} and terms s, t , one can check in polynomial time whether $\mathcal{E} \vdash s \simeq t$.*

Proof. See [8] or [14]. (In fact, this problem is P-complete.) \square

We write $\mathcal{E}' \vdash \mathcal{E}$ if for every equation $(s \simeq t) \in \mathcal{E}$ we have $\mathcal{E}' \vdash s \simeq t$. We call two sets of equations \mathcal{E} and \mathcal{E}' *equivalent*, denoted $\mathcal{E} \equiv \mathcal{E}'$, if $\mathcal{E} \vdash \mathcal{E}'$ and $\mathcal{E}' \vdash \mathcal{E}$.

In the sequel we shall use the following two lemmas.

Lemma 2.2 (Lemma on constants). *Let \mathcal{E} and \mathcal{E}' be sets of equations. For every constant c and term t , if $\mathcal{E} \vdash \mathcal{E}'$, then $\mathcal{E}_c^t \vdash \mathcal{E}'_c^t$.*

Proof. Standard. \square

Lemma 2.3. *Let \mathcal{E} be a set of ground equations, E be a ground equation, c be a constant, and t be a ground term in which c does not occur. Then $\mathcal{E} \cup \{c \simeq t\} \vdash E$ if and only if $\mathcal{E}_c^t \vdash E_c^t$.*

Proof. Immediate by Lemma 2.2. \square

2.2. Words and finite automata

This section defines *words* and *finite automata*. We shall also introduce a notation for monadic terms, which allows us to easily come from terms to words and back.

Let \mathcal{F} be a finite non-empty set, called the *alphabet*. Its elements are called *letters*. *Words* are finite sequences of letters. We denote words by a juxtaposition of its letters, as

$$W = a_1 a_2 \dots a_n.$$

The natural number n is called the *length* of the word W and denoted $|W|$. We denote by ε the *empty word*, that is the unique word of length zero. The set of all words with letters in \mathcal{F} is denoted by \mathcal{F}^* .

It will be convenient for us to use the alphabet \mathcal{F} also as the set of unary function symbols of a monadic signature $\mathcal{F} \cup \mathcal{C}$. Every term s in such a signature has the form $f_1(f_2(\dots f_n(t)\dots))$ where $n \geq 0$, f_1, \dots, f_n are unary function symbols and t is a constant or a variable. We shall denote such a term s in the reversed Polish notation, i.e. as $t f_n \dots f_2 f_1$. Thus, every term can be represented in the form tW , where t is a constant or a variable and W is a word. Similarly, any term of the form $f_1(f_2(\dots f_n(t)\dots))$, where t is an arbitrary term, will be written as $t f_n \dots f_2 f_1$.

A *finite automaton* \mathcal{A} on the alphabet \mathcal{F} is a quadruple (Q, I, T, E) , where Q is a finite set, called the *set of states*, I and T are distinguished subsets of Q , called the sets of *initial* and *terminal* states, respectively. The set $E \subseteq Q \times \mathcal{F} \times Q$ is the *set of edges* of \mathcal{A} . An edge (p, f, q) is also denoted $p \xrightarrow{f} q$. The automaton is called *deterministic*, or DFA for short, if whenever $(p, f, q_1) \in E$ and $(p, f, q_2) \in E$, then $q_1 = q_2$.

We say a word $f_1 \dots f_n$ is *recognized* by an automaton (Q, I, T, E) if there is a sequence of states $q_0 \dots q_n$ such that $q_0 \in I$, $q_n \in T$, and $q_{i-1} \xrightarrow{f_i} q_i$ for all $i \in \{1, \dots, n\}$. A set of words is called *regular* if it is the set of words recognized by some automaton.

The *DFA intersection nonemptiness problem* is the following decision problem. Given any finite set $\{\mathcal{A}_1, \dots, \mathcal{A}_n\}$ of DFAs, does there exist a word recognized by each automaton in this set. The following statement is proved in [9]:

Lemma 2.4 (DFA intersection nonemptiness problem). *The DFA intersection non-emptiness problem is PSPACE-complete.*

2.3. Word equations

In addition to the alphabet \mathcal{F} , we shall also consider a countable set \mathcal{V} of *word variables*, denoted u, v, w . A *word equation* is any expression of the form $V \simeq W$, where $V, W \in (\mathcal{F} \cup \mathcal{V})^*$. A *word substitution* is any expression $\sigma = \{V_1/v_1, \dots, V_n/v_n\}$, where v_i are word variables and V_i are words in \mathcal{F}^* . Its *domain*, denoted $\text{dom}\sigma$ is the set $\{v_1, \dots, v_n\}$. The *application* of such a word substitution σ to a word $W \in (\mathcal{F} \cup \mathcal{V})^*$, is

the operation of simultaneous replacement of all occurrences of v_i by V_i . The result of the application is the word denoted $W\sigma$. A word substitution σ is called a *solution to a word equation* $U \simeq V$ if all variables in U, V belong to $\text{dom}(\sigma)$ and we have $U\sigma = V\sigma$. A *system of word equations* is any finite set of word equations, its *solution* is any substitution solving all equations in the system. Words will be denoted by U, V, W and word substitutions by σ . We say that a word substitution σ' *extends* a word substitution σ if (i) $\text{dom}(\sigma) \subseteq \text{dom}(\sigma')$, and (ii) for every $v \in \text{dom}(\sigma)$ we have $v\sigma = v\sigma'$.

Makanin [10] proves that word equations are decidable. Analyzing Makanin's algorithm, Schultz [13] proves the following result.

Lemma 2.5 (Decidability of word equations with regular constraints). *The problem of solvability of word equations where every word variable u_i ranges over a regular set S_i , is decidable.*

It is known that the problem of solvability of word equations is NP-hard. The best known lower bound is 3-NEXP [6, 7].

2.4. Equational logic and rigid equations

Let \mathcal{R} be a system of rigid equations. The *signature of \mathcal{R}* is defined as the signature consisting of all constants and function symbols occurring in \mathcal{R} ; and in addition a fixed constant if \mathcal{R} contains no constants. A solution θ to \mathcal{R} is called *grounding for \mathcal{R}* if for every variable x occurring in \mathcal{R} the term $x\theta$ is ground. A substitution θ is called *relevant for \mathcal{R}* if all terms in $\text{ran}(\theta)$ are terms in the signature of \mathcal{R} .

In the sequel, we shall need the following technical property of systems of rigid equations. We omit the straightforward proof.

Lemma 2.6 (Existence of relevant grounding solutions). *Let \mathcal{R} be a solvable system of rigid equations. Then there exists a solution θ to \mathcal{R} that is grounding and relevant for \mathcal{R} .*

We shall introduce one particular kind of rigid equations that will be used as a technical tool for proofs in this paper. For every monadic signature $\mathcal{F} \cup \mathcal{C}$, any variable x , and any constant $c \in C$ introduce the following rigid equation:

$$\text{Gr}_{\mathcal{F} \cup \mathcal{C}}(x) \triangleq \{d \simeq c \mid d \in \mathcal{C}\} \cup \{cf \simeq c \mid f \in \mathcal{F}\} \vdash_{\forall} x \simeq c.$$

We shall use the following obvious lemma:

Lemma 2.7. *A substitution θ is a solution to $\text{Gr}_{\mathcal{F} \cup \mathcal{C}}(x)$ if and only if $x\theta \in T_{\mathcal{F} \cup \mathcal{C}}$.*

As a consequence, we have

Lemma 2.8. *For every system \mathcal{R} of rigid equations there is a system \mathcal{R}' of rigid equations such that for every substitution θ , θ is a solution to \mathcal{R}' if and only if θ*

is a grounding relevant solution to \mathcal{R} . In addition, (i) \mathcal{R}' can be found by \mathcal{R} in polynomial time; and (ii) \mathcal{R}' has ground left sides if \mathcal{R} has ground left sides.

Proof. Let x_1, \dots, x_n be all variables in \mathcal{R} and $\mathcal{F} \cup \mathcal{C}$ be the signature of \mathcal{R} . Define $\mathcal{R}' \Leftarrow \mathcal{R} \cup \{Gr_{\mathcal{F} \cup \mathcal{C}}(x_i) \mid i \in \{1, \dots, n\}\}$. Then apply Lemma 2.7. \square

The following lemma says that we can essentially consider signatures with only one constant.

Lemma 2.9. *Given a system \mathcal{R} of rigid equations in the signature $\mathcal{F} \cup \mathcal{C}$, one can find in polynomial time an equi-solvable system \mathcal{R}' of rigid equations in a signature $\mathcal{F}' \cup \{c\}$.*

Proof. Let c_1, \dots, c_n be all constants occurring in \mathcal{R} . Take n new function symbols f_1, \dots, f_n and define \mathcal{F}' as $\mathcal{F} \cup \{f_1, \dots, f_n\}$. Transform \mathcal{R} into \mathcal{R}' as follows. First, replace all occurrences of constants c_i by terms cf_i . Second, for every variable x occurring in \mathcal{R} add the following rigid equation

$$\{cf_i g \simeq cf_i \mid i = 1, \dots, n, g \in \mathcal{F}\} \cup \{cf_i \simeq cf_i \mid i = 2, \dots, n\} \vdash_{\forall} x \simeq cf_1.$$

It is clear that solutions of this rigid equation are the substitutions θ such that $x\theta = cf_i W$, where W is a word on \mathcal{F} . Then, it is not hard to argue that a substitution

$$\{c_{i_1} W_1/x_1, \dots, c_{i_m} W_m/x_m\}$$

is a solution to \mathcal{R} if and only if the substitution

$$\{cf_{i_1} W_1/x_1, \dots, cf_{i_m} W_m/x_m\}$$

is a solution to \mathcal{R}' . \square

2.5. Rewrite rules

This section introduces a technique standard in the theory of ground systems of rewrite rules. However, we shall use ordinary equations instead of rewrite rules.

Introduce an ordering \succ on terms in $T_{\mathcal{F} \cup \mathcal{C}}$ in the following way. Let $>$ be any total ordering on $\mathcal{F} \cup \mathcal{C}$ and $s = cf_1 \dots f_m$, $t = dg_1 \dots g_n$. Then $s \succ t$ if one of the following conditions is true:

1. $m > n$;
2. $m = n$ and the string $cf_1 \dots f_m$ is greater than $dg_1 \dots g_n$ in the lexicographic ordering induced by $>$.

The ordering \succ is total, well-founded, and can be extended to a simplification ordering [5]. Some properties of the ordering formulated below are simple consequence of standard statements in the theory of rewrite systems. Their proofs may be found in, e.g., [5]. Note that the ordering \succ depends on the ordering $>$. In the definitions below we assume that we have chosen a fixed ordering $>$ on $\mathcal{F} \cup \mathcal{C}$, and hence \succ is also fixed.

Let $\mathcal{E}, \mathcal{E}'$ be finite sets of ground equations and \mathcal{E} contain distinct equations $s \simeq t$ and $r[s] \simeq u$. We say that \mathcal{E}' is obtained from \mathcal{E} by simplification from $s \simeq t$ into $r[s] \simeq u$, denoted $\mathcal{E} \rightarrow \mathcal{E}'$ if

$$\mathcal{E}' = (\mathcal{E} \setminus \{r[s] \simeq u\}) \cup \{r[t] \simeq u\}.$$

The reflexive and transitive closure of the relation \rightarrow on sets of ground equations is denoted by \rightarrow^* . A set of equations \mathcal{E} is called *irreducible* if there exists no \mathcal{E}' such that $\mathcal{E} \rightarrow \mathcal{E}'$. We shall use the following statements, which are easy to prove.

Lemma 2.10. *Let \mathcal{E} be a finite set of equations. Then there exists an irreducible set of equations \mathcal{E}' such that $\mathcal{E} \rightarrow^* \mathcal{E}'$.*

Lemma 2.11. *Let $\mathcal{E} \rightarrow^* \mathcal{E}'$. Then $\mathcal{E} \equiv \mathcal{E}'$.*

Let \mathcal{E} be an irreducible set of ground equations. We write $t \rightarrow_{\mathcal{E}} t'$ if there exists an equation $(r \simeq s) \in \mathcal{E}$ such that $r \succ s$, and t' is obtained from t by the replacement of one occurrence of the subterm r by s . The relation $\rightarrow_{\mathcal{E}}^*$ is the reflexive and transitive closure of $\rightarrow_{\mathcal{E}}$. A term t is called *irreducible with respect to \mathcal{E}* if there is no term s such that $t \rightarrow_{\mathcal{E}} s$. The *normal form of a term t w.r.t. \mathcal{E}* , denoted $t \downarrow_{\mathcal{E}}$, is the term s such that $t \rightarrow_{\mathcal{E}}^* s$ and s is irreducible w.r.t. \mathcal{E} . The normal form of any term exists and is unique. We shall use the following statements, which are easy to prove.

Lemma 2.12. *Let \mathcal{E} be an irreducible set of ground equations and s, t be terms. Then $\mathcal{E} \vdash s \simeq t$ if and only if $s \downarrow_{\mathcal{E}} = t \downarrow_{\mathcal{E}}$.*

Lemma 2.13. *Let \mathcal{E} be an irreducible set of ground equations and $s \downarrow_{\mathcal{E}} = t$. Then for each subterm t' of t there is a term $r \in T(\mathcal{E} \cup \{s \simeq s\})$ such that $\mathcal{E} \vdash t' \simeq r$.*

Lemma 2.14. *Let $\mathcal{E} \rightarrow^* \mathcal{E}'$. Then for each term $t' \in T(\mathcal{E}')$ there is a term $t \in T(\mathcal{E})$ such that $\mathcal{E} \vdash t' \simeq t$.*

2.6. Mixing words and rigid equations

We call a *word term*, or simply *w-term*, in the signature $\mathcal{F} \cup \mathcal{C}$ any expression of the form cW such that $c \in \mathcal{C}$ and $W \in (\mathcal{F} \cup \mathcal{V})^*$. A *w-equation* is any expression $cV \simeq dW$, where cV and dW are w-terms. A *rigid w-equation* is any expression of the form $\mathcal{W} \vdash_{\forall} cV \simeq dW$, where \mathcal{W} is a finite set of w-equations and cV and dW are w-terms. A *system of rigid w-equations* is any finite set of rigid w-equations. The *signature of a system of rigid w-equations* is defined similar to that of a system of rigid equations. Sets of w-equations will be denoted by \mathcal{W} , while sets of rigid w-equations by \mathcal{S} .

A *solution to a rigid w-equation* $\mathcal{W} \vdash_{\forall} cV \simeq dW$ is any word substitution σ whose domain contains all word variables in \mathcal{W}, V, W such that $\mathcal{W}\sigma \vdash cV\sigma \simeq dW\sigma$. A

solution to a system \mathcal{S} of rigid w-equations is any word substitution that is a solution to every rigid w-equation in \mathcal{S} .

Note that a ground w-equation is also an ordinary equation.

Relevant solutions to systems of rigid w-equations are defined similar to those of systems of rigid equations. Similar to Lemma 2.6, we have the following property:

Lemma 2.15. *Let \mathcal{S} be a solvable system of rigid w-equations. Then there exists a solution σ to \mathcal{S} relevant for \mathcal{S} .*

In Lemma 2.17 below we show that one can consider systems of rigid w-equations instead of systems of rigid equations. First, we shall prove a technical lemma.

Lemma 2.16. *Given any system \mathcal{S} of rigid w-equations, one can find in polynomial time an equi-solvable system \mathcal{S}' of rigid w-equations such that for every rigid w-equation $S \in \mathcal{S}'$ we have*

1. *each word variable has at most one occurrence in S ;*
2. *one of the following conditions hold:*
 - (a) *S has the form $\vdash_{\forall} cu \simeq cvw$, where u, v, w are pairwise distinct word variables;*
or
 - (b) *for every w-term cW occurring in S , we have $|W| \leq 1$.*

Proof. First, we show how to transform \mathcal{S} to make every word variable have at most one occurrence in any rigid w-equation in \mathcal{S} . Let a word variable u has more than one occurrence in a rigid w-equation $S \in \mathcal{S}$. Replace one occurrence of u by a new word variable v and add to \mathcal{S} the rigid w-equation $\vdash_{\forall} cu \simeq cv$. Evidently, the resulting system and the system \mathcal{S} are equi-solvable.

Then we show how to reduce the length of words occurring in \mathcal{S} by adding to \mathcal{S} rigid w-equations $\vdash_{\forall} cu \simeq cvw$. Let cVW be any word occurring in \mathcal{S} . Introduce three new word variables u, v, w and do the following. Replace this occurrence of cVW by cu and add to \mathcal{S} rigid w-equations $\vdash_{\forall} cv \simeq cV$, $\vdash_{\forall} cw \simeq cW$ and $\vdash_{\forall} cu \simeq cvw$. Evidently, the resulting system and \mathcal{S} are equi-solvable.

It is easy to prove that after several such transformations of \mathcal{S} we obtain, in polynomial time, a system \mathcal{S}' satisfying the conditions. \square

Lemma 2.17. *The problem of solvability of systems of rigid w-equations is polynomial-time equivalent to monadic SREU.*

Proof.

1. Let \mathcal{S} be a system of rigid w-equations and $\mathcal{F} \cup \mathcal{C}$ be its signature. By Lemma 2.16, we can assume that all rigid w-equations in \mathcal{S} satisfy the conditions of that lemma. For every variable v occurring in \mathcal{S} , introduce a new constant c_v . Define a system of rigid equations \mathcal{R} in the following way. First, we make a system of rigid equations \mathcal{R}' out of \mathcal{S} by the following transformations applied to every rigid w-equation S in \mathcal{S} .

- (a) If S has the form $\vdash_{\forall} cu \simeq cvw$, where u, v, w are pairwise distinct word variables. Then replace S by the rigid equation $c_u \simeq c_v, c_w \simeq v \vdash_{\forall} w \simeq u$.
- (b) If for every w -term cW occurring in S , we have $|W| \leq 1$. For every w -term of the form cu occurring in S , replace it by u . Let the resulting rigid equation have the form $\mathcal{E} \vdash_{\forall} E$. Then add to \mathcal{E} equations $c \simeq c_u$ for every w -term cu occurring in S .

Second, add to \mathcal{R}' rigid equations $Gr_{(\mathcal{F}, \{c_u\})}(u)$ for every word variable u occurring in \mathcal{S} . We prove that the resulting system of rigid equations \mathcal{R} and the system \mathcal{S} are equi-solvable.

Note that because of rigid equations $Gr_{(\mathcal{F}, \{c_u\})}(u)$, every solution to \mathcal{R} has the form $\{c_u W_u/u, \dots, c_v W_v/v\}$, where u, \dots, v are all variables occurring in \mathcal{S} and $W_u, \dots, W_v \in \mathcal{F}^*$.

By Lemmas 2.6 and 2.15, we can restrict ourselves to relevant solutions. Now we shall prove that for every word substitution $\sigma = \{W_u/u, \dots, W_v/v\}$ grounding and relevant for \mathcal{S} , σ is solution to \mathcal{S} if and only if the substitution $\theta = \{c_u W_u/u, \dots, c_v W_v/v\}$ is a solution to \mathcal{R} .

- (a) Consider rigid w -equations $s \in \mathcal{S}$ of the form $\vdash_{\forall} cu \simeq cvw$. In this case \mathcal{R} contains the equation $r = (c_u \simeq c_v, c_w \simeq v \vdash_{\forall} w \simeq u)$. Obviously, σ is a solution to s if and only if $W_u = W_v W_w$.

Now consider now when θ is a solution to $c_u \simeq c_v, c_w \simeq v \vdash_{\forall} w \simeq u$. It is a solution if and only if $c_u \simeq c_v, c_w \simeq c_v W_v \vdash_{\forall} c_w W_w \simeq c_u W_u$. Applying Lemma 2.3 twice, we can show that this is equivalent to $\vdash_{\forall} c_v W_v W_w \simeq c_v W_u$, i.e. to $W_v W_w = W_u$.

- (b) Now consider the case of rigid w -equations $S \in \mathcal{S}$ such that for every w -term cW occurring in S , we have $|W| \leq 1$. We demonstrate this case by an example. For instance, let S have the form $au \simeq cv, a \simeq aw \vdash_{\forall} az \simeq cf$. Then \mathcal{R} contains the rigid equation $R = (a \simeq c_u, c \simeq c_v, a \simeq c_w, a \simeq c_z, u \simeq v, a \simeq w \vdash_{\forall} z \simeq cf)$. The word substitution σ is a solution to S if and only if

$$aW_u \simeq cW_v, a \simeq aW_w \vdash_{\forall} aW_z \simeq cf.$$

The substitution θ is a solution to R if and only if

$$a \simeq c_u, c \simeq c_v, a \simeq c_w, a \simeq c_z, c_u W_u \simeq c_v W_v, a \simeq c_w W_w \vdash_{\forall} c_z W_z \simeq cf.$$

Applying Lemma 2.3, we see that these conditions are equivalent (the key point in applying this lemma is that every word variable occurs in \mathcal{S} at most once).

It is clear that the reduction is polynomial.

2. Let \mathcal{R} be a system of rigid equations. By Lemma 2.9 we can assume that the signature of \mathcal{R} contains one constant c . Transform \mathcal{R} into a system of rigid w -equations \mathcal{S} by replacing each occurrence of every variable x by the w -term cx . Applying Lemmas 2.6 and 2.15 on the existence of relevant and grounding solutions, we can show that \mathcal{R} is solvable if and only if \mathcal{S} is solvable. \square

Using this lemma and the same idea as in Lemma 2.9, we can restrict ourselves to rigid w-equations with one constant.

Lemma 2.18. *Monadic SREU is polynomial-time equivalent the problem of solvability of systems of rigid w-equations with one constant.*

3. Ground left sides

In this section we prove that monadic SREU with ground left sides is decidable and PSPACE-hard.

3.1. SREU with ground left sides is PSPACE-hard

Lemma 3.1. *Let $\mathcal{A} = (Q, I, T, E)$ be a DFA over \mathcal{F} . There exists a system \mathcal{R} of two monadic rigid equations of one variable x with the following properties:*

1. \mathcal{R} has ground left sides;
2. for every solution θ to \mathcal{R} we have $x\theta = cW$, where $W \in \mathcal{F}^*$ and c is a fixed constant;
3. for every word $W \in \mathcal{F}^*$, the substitution $\{cW/x\}$ is a solution to \mathcal{R} if and only if W is recognized by \mathcal{A} .

In addition, \mathcal{R} can be constructed from \mathcal{A} in polynomial time.

Proof. Without loss of generality, we can assume that I consists of one state (see, e.g., [11]). By renaming states, we can assume that $I = \{c\}$. Let F be a unary function symbol not in \mathcal{F} and d be a constant not in Q . Define \mathcal{R} as $\{R_1, R_2\}$, where

$$\begin{aligned} R_1 &= \{pf \simeq q \mid (p \xrightarrow{f} q) \in E\} \cup \{rF \simeq d \mid r \in T\} \vdash_{\forall} xF \simeq d, \\ R_2 &= Gr_{(\mathcal{F}, \{c\})}(x). \end{aligned}$$

Consider any substitution $\theta = \{t/x\}$. By Lemma 2.7, θ is a solution to R_2 if and only if t has the form cW such that $W \in \mathcal{F}^*$. Consider when such substitution $\{cW/x\}$ is also a solution to R_1 . By definition, this means

$$\{pf \simeq q \mid (p \xrightarrow{f} q) \in E\} \cup \{rF \simeq d \mid r \in T\} \vdash cWF \simeq d. \quad (1)$$

Since the automaton is deterministic, the left side of (1) is irreducible. Using Lemma 2.12, one can see that (1) holds if and only if W is recognizable by \mathcal{A} . Evidently, \mathcal{R} is constructed by \mathcal{A} in polynomial time. \square

Lemma 3.2. *The DFA intersection nonemptiness problem is polynomial-time reducible to monadic SREU with one variable and ground left sides.*

Proof. Let $\mathcal{A}_1, \dots, \mathcal{A}_n$ be DFAs. Let \mathcal{R}_i , where $i \in \{1, \dots, n\}$ be the system of rigid equations constructed from \mathcal{A}_i as in Lemma 3.1. Define $\mathcal{R} = \bigcup_{i=1}^n \mathcal{R}_i$. By Lemma 3.1,

every solution to \mathcal{R} has the form $\{cW/x\}$ and any substitution $\{cW/x\}$ is a solution to \mathcal{R} if and only if W is recognized by each \mathcal{A}_i . Hence, \mathcal{R} is solvable if and only if there is a word recognizable by all \mathcal{A}_i . Evidently, \mathcal{R} is constructed by $\mathcal{A}_1, \dots, \mathcal{A}_n$ in polynomial time. \square

Combining Lemmas 2.4 and 3.2 we obtain

Theorem 1. *Monadic SREU with one variable and ground left sides is PSPACE-hard.*

3.2. Monadic SREU with ground left sides is decidable

A finite set \mathcal{E} of equations is in automaton form if

1. every equation in \mathcal{E} has the form $cf \simeq d$;
2. for every two w-equations $cf \simeq d_1$ and $cf \simeq d_2$ in \mathcal{E} we have $d_1 = d_2$;

Note that any set of equations in automaton form is irreducible.

Lemma 3.3. *Given any rigid w-equation S with a ground left side, one can effectively find in polynomial time a rigid w-equation S' with a ground left side such that*

1. S and S' have the same solutions;
2. the left side of S' is in automaton form;
3. the right side of S' does not contain subterms of the form cf .

Proof. We define a series of equivalence-preserving transformations of the current w-equation S that produces a w-equation whose left side is in the automaton form. For the reader's convenience, we illustrate the process on the example where the initial S is $ahh \simeq a, ag \simeq b \vdash_{\forall} bxg \simeq bhyg$.

Let T be the set of all ground terms occurring in S and their subterms. In our example $T = \{ahh, ah, a, ag, b, bh\}$. Introduce a set of new constants $C_T = \{c_t \mid t \in T\}$.

1. Replace each constant d occurring in S by c_d . In our example, S becomes

$$c_a h h \simeq c_a, c_a g \simeq c_b \vdash_{\forall} c_b x g \simeq c_b h y g.$$

Obviously, this transformation does not change the set of solutions.

2. Add to the left side of S the set of equations $\{c_t f \simeq c_{tf} \mid tf \in T\}$. In our example, S becomes

$$\begin{aligned} c_a h h \simeq c_{ahh}, c_a h \simeq c_{ah}, c_a g \simeq c_{ag}, c_b h \simeq c_{bh}, c_a h h \simeq c_a, c_a g \simeq c_b \\ \vdash_{\forall} c_b x g \simeq c_b h y g. \end{aligned}$$

Applying Lemma 2.3, one can show that the set of solutions to S does not change.

3. Get rid of all ground non-constant terms in the right side by repeatedly replacing terms of the form $c_{tf} f$ by c_{tf} . In our example, S becomes

$$\begin{aligned} c_a h h \simeq c_{ahh}, c_a h \simeq c_{ah}, c_a g \simeq c_{ag}, c_b h \simeq c_{bh}, c_a h h \simeq c_a, c_a g \simeq c_b \\ \vdash_{\forall} c_b x g \simeq c_{bh} y g. \end{aligned}$$

Obviously, the set of solutions does not change since the left side contains the equations of the form $c_t f \simeq c_{t'f}$.

4. Let \mathcal{E} be the left side of S . Replace \mathcal{E} by an irreducible set \mathcal{E}' such that $\mathcal{E} \rightarrow^* \mathcal{E}'$ (such \mathcal{E}' exists by Lemma 2.10). In our example, S becomes

$$\begin{aligned} c_{ah}h &\simeq c_{ahh}, c_a h \simeq c_{ah}, c_a g \simeq c_{ag}, c_b h \simeq c_{bh}, c_{ahh} \simeq c_a, c_{ag} \simeq c_b \\ \vdash_{\forall} c_b x g &\simeq c_{bh} y g. \end{aligned}$$

It is easy to prove that the left side of \mathcal{E}' consists of equations of the form $c f \simeq d$ or $c \simeq d$. By Lemma 2.11, $\mathcal{E} \equiv \mathcal{E}'$. Hence, the set of solutions does not change.

5. Get rid of all equations of the form $c \simeq d$ in the left side of S by removing them and replacing c by d in S . In our example, S becomes

$$c_{ah}h \simeq c_a, c_a h \simeq c_{ah}, c_a g \simeq c_b, c_b h \simeq c_{bh}, \vdash_{\forall} c_b x g \simeq c_{bh} y g.$$

By Lemma 2.2, the set of solutions does not change. Since the left side of S is irreducible, it is in automaton form.

Evidently, S' is constructed by S in polynomial time. \square

Let \mathcal{E} be a set of equations in automaton form and c, d be any constants. Denote by $\mathcal{A}(\mathcal{E}, c, d)$ the following DFA (Q, I, T, E) . Its alphabet is the set of function symbols occurring in \mathcal{E} . The set of states Q is the set of all constants occurring in \mathcal{E}, c, d . The sets of initial states and terminal states are defined by $I \Leftarrow \{c\}$ and $T \Leftarrow \{d\}$. Finally, the set of edges is defined by

$$E \Leftarrow \{a \xrightarrow{f} b \mid (af \simeq b) \in \mathcal{E}\}.$$

Lemma 3.4. *A word W is recognized by $\mathcal{A}(\mathcal{E}, c, d)$ if and only if $\mathcal{E} \vdash cW \simeq d$.*

Proof. Immediate by Lemma 2.12. \square

Lemma 3.5. *Let \mathcal{E} be a set of equations in automaton form, $W, W' \in \mathcal{F}^*$, and c, c' be constants. Then $\mathcal{E} \vdash cW \simeq c'W'$ if and only if there is a constant d and words U, U', V such that $W = UV$, $W' = U'V$, U is recognized by $\mathcal{A}(\mathcal{E}, c, d)$ and U' is recognized by $\mathcal{A}(\mathcal{E}, c', d)$.*

Proof. (\Rightarrow) We have $\mathcal{E} \vdash cW \simeq c'W'$. By Lemma 2.12 we have $cW \downarrow_{\mathcal{E}} = c'W' \downarrow_{\mathcal{E}}$. Choose d and V such that $cW \downarrow_{\mathcal{E}} = dV$. Define U and U' such that $W = UV$ and $W' = U'V$. We have $\mathcal{E} \vdash cU \simeq d$ and $\mathcal{E} \vdash c'U' \simeq d$. By Lemma 3.4 words U and U' are recognized by $\mathcal{A}(\mathcal{E}, c, d)$ and $\mathcal{A}(\mathcal{E}, c', d)$, respectively.

(\Leftarrow) We have $W = UV$, $W' = U'V$, U is recognized by $\mathcal{A}(\mathcal{E}, c, d)$ and U' is recognized by $\mathcal{A}(\mathcal{E}, c', d)$. By Lemma 3.4, we have $\mathcal{E} \vdash cU \simeq d$ and $\mathcal{E} \vdash c'U' \simeq d$. Hence, $\mathcal{E} \vdash cUV \simeq dV$ and $\mathcal{E} \vdash c'U'V \simeq dV$. Then $\mathcal{E} \vdash cUV \simeq c'U'V$, i.e. $\mathcal{E} \vdash cW \simeq c'W'$. \square

Let R_0, R_1, \dots be an enumeration of all regular sets of words on \mathcal{F} . We call a *regular constraint* any expression $R_i(U)$ where U is a word on $\mathcal{F} \cup \mathcal{V}$. A solution to such regular constraint is any word substitution σ such that $U\sigma \in R_i$.

Lemma 3.6. *The problem of solvability of systems of rigid w-equations with ground left sides effectively reduces to word equations with regular constraints.*

Proof. Let $\mathcal{S} = \{S_1, \dots, S_n\}$ be such a system of rigid w-equations. By Lemma 3.3, we can assume that the left sides of all S_i are in automaton form. Let $S_i = (\mathcal{E}_i \vdash_{\forall} c_i W_i \simeq c'_i W'_i)$, for all $i \in \{1, \dots, n\}$. Let $u_1, \dots, u_n, v_1, \dots, v_n$ and u'_1, \dots, u'_n be word variables not in \mathcal{S} . By Lemma 3.5, the system \mathcal{S} is solvable if and only if there are constants d_i occurring in S_i , for all $i \in \{1, \dots, n\}$ such that the following system of word equations and regular constraints is solvable:

$$\begin{aligned} W_1 &\simeq u_1 v_1 \quad u_1 \text{ is recognized by } \mathcal{A}(\mathcal{E}_1, c_1, d_1), \\ \dots &\quad \dots \\ W_n &\simeq u_n v_n \quad u_n \text{ is recognized by } \mathcal{A}(\mathcal{E}_n, c_n, d_n), \\ W'_1 &\simeq u'_1 v_1 \quad u'_1 \text{ is recognized by } \mathcal{A}(\mathcal{E}_1, c'_1, d_1), \\ \dots &\quad \dots \\ W'_n &\simeq u'_n v_n \quad u'_n \text{ is recognized by } \mathcal{A}(\mathcal{E}_n, c'_n, d_n). \end{aligned}$$

To conclude the proof we note that there is only a finite number of choices for d_i . \square

Theorem 2. *Monadic SREU with ground left sides is decidable.*

Proof. By Lemma 2.17, monadic SREU with ground left sides is effectively reducible to the problem of solvability of systems of rigid w-equations. By Lemma 3.6 the latter problem is effectively reducible to word equations with regular constraints. Then apply Lemma 2.5. \square

4. One-variable case

In this section we prove that monadic SREU with one variable is PSPACE-complete. We assume that all rigid equations are of one variable x . We shall write $\mathcal{E}(x)$ to denote all occurrences of a variable x in \mathcal{E} , and write $\mathcal{E}(t)$ to denote the set of equations obtained from \mathcal{E} by replacement of all occurrences of x by t . We shall use similar notation for terms, for example, $s(x)$. Using this notation, we can write any rigid equation of one variable x as $\mathcal{E}(x) \vdash_{\forall} s(x) \simeq t(x)$.

Lemma 4.1. *Let $\mathcal{E}(x)$ be a finite set of equations of one variable x and $s(x), t(x)$ be terms of one variable x such that $\mathcal{E}(x) \not\vdash s(x) \simeq t(x)$. Let c be a constant not occurring in $\mathcal{E}(x), s(x), t(x)$ and r be a ground term such that c does not occur in r . If $\mathcal{E}(r) \vdash s(r) \simeq t(r)$, then there exists a ground term $r' \in T(\mathcal{E}(c) \cup \{s(c) \simeq t(c)\})$ such that $\mathcal{E}(c) \vdash r \simeq r'$.*

Proof. Suppose $\mathcal{E}(r) \vdash s(r) \simeq t(r)$. Hence, $r \simeq c, \mathcal{E}(r) \vdash s(r) \simeq t(r)$. This implies $r \simeq c, \mathcal{E}(c) \vdash s(c) \simeq t(c)$. Consider any ordering \succ in which c is the least element. By Lemma 2.10, there exists an irreducible \mathcal{E}' such that $\mathcal{E}(c) \rightarrow^* \mathcal{E}'$. Let s' and t' be the normal forms of $s(c)$ and $t(c)$, respectively, w.r.t. \mathcal{E}' . Let us prove that $s' \neq t'$. Suppose, by contradiction, $s' = t'$. By Lemma 2.12, we have $\mathcal{E}' \vdash s(c) \simeq t(c)$. By Lemma 2.11, we have $\mathcal{E}(c) \vdash s(c) \simeq t(c)$. By Lemma 2.2, we have $\mathcal{E}(x) \vdash s(x) \simeq t(x)$, a contradiction.

Evidently, we have $r \simeq c, \mathcal{E}' \vdash s' \simeq t'$. Consider two cases:

1. The set of equations $\{r \simeq c\} \cup \mathcal{E}'$ is reducible. Since \mathcal{E}' is irreducible, we have $r \in T(\mathcal{E}')$. By Lemma 2.14, there is $r' \in T(\mathcal{E}(c))$ such that $\mathcal{E}(c) \vdash r \simeq r'$.
2. The set of equations $\{r \simeq c\} \cup \mathcal{E}'$ is irreducible. By Lemma 2.12, normal forms of s' and t' w.r.t. $\{r \simeq c\} \cup \mathcal{E}'$ coincide. Since $s' \neq t'$, one of the terms s', t' , for example s' , is different from its normal form w.r.t. $\{r \simeq c\} \cup \mathcal{E}'$. Since s' is irreducible w.r.t. \mathcal{E}' , the term r is a subterm of s' . By Lemma 2.13, there is a subterm r' of $s(c)$ such that $\mathcal{E}(c) \vdash r \simeq r'$. \square

Lemma 4.2. *Let $\mathcal{E}(x) \vdash_{\forall} s(x) \simeq t(x)$ be a rigid equation of one variable x , c be a constant not occurring in this rigid equation, r be a ground term in which c does not occur and $\mathcal{E}(x) \not\vdash s(x) \simeq t(x)$. Then the substitution $\theta = \{r/x\}$ is a solution to this rigid equation if and only if there is a ground term $r' \in T(\mathcal{E}(c) \cup \{s(c) \simeq t(c)\})$ such that $\mathcal{E}(c), \mathcal{E}(r') \vdash s(r') \simeq t(r')$ and θ is a solution to $\mathcal{E}(c) \vdash_{\forall} r' \simeq x$.*

Proof. (\Rightarrow) We have that θ is a solution to $\mathcal{E}(x) \vdash_{\forall} s(x) \simeq t(x)$. Then $\mathcal{E}(r) \vdash s(r) \simeq t(r)$. By Lemma 4.1, there is a term $r' \in T(\mathcal{E}(c) \cup \{s(c) \simeq t(c)\})$ such that $\mathcal{E}(c) \vdash r \simeq r'$. Then $\mathcal{E}(r'), \mathcal{E}(c) \vdash s(r') \simeq t(r')$.

(\Leftarrow) We have $\mathcal{E}(c), \mathcal{E}(r') \vdash s(r') \simeq t(r')$ and $\mathcal{E}(c) \vdash_{\forall} r' \simeq x$. Then $\mathcal{E}(c), \mathcal{E}(r) \vdash s(r) \simeq t(r)$. By Lemma 2.2, we can substitute r for c obtaining $\mathcal{E}(r) \vdash s(r) \simeq t(r)$. \square

Lemmas 4.1 and 4.2 also hold for non-monadic signatures [1].

Lemma 4.3. *Monadic SREU with one variable is in PSPACE.*

Proof. We give a nondeterministic polynomial-time algorithm, which uses an oracle for DFA intersection; it follows (by Kozen's [9] and Savich's theorems) that the problem is in PSPACE.

Let \mathcal{R} be a system of rigid equations of one variable x whose signature is $\mathcal{F} \cup \mathcal{C}$. It has the form

$$\begin{aligned} \mathcal{E}_1 \vdash_{\forall} s_1(x) &\simeq t_1(x), \\ \dots \\ \mathcal{E}_n \vdash_{\forall} s_n(x) &\simeq t_n(x). \end{aligned}$$

By Lemma 2.6, we can restrict ourselves to relevant grounding solutions $\theta = \{r/x\}$ only. Let c be a variable not in $\mathcal{F} \cup \mathcal{C}$. By Lemma 4.2, θ is a solution to \mathcal{R} if and only if there are ground terms $r'_i \in T(\mathcal{E}_i(c) \cup \{s_i(c) \simeq t_i(c)\})$, where $i \in \{1, \dots, n\}$ such that $\mathcal{E}(c), \mathcal{E}(r') \vdash s(r') \simeq t(r')$ and θ is a solution to the system

$$\begin{aligned} \mathcal{E}_1(c) &\vdash_{\forall} r'_1 \simeq x, \\ &\dots \\ \mathcal{E}_n(c) &\vdash_{\forall} r'_n \simeq x. \end{aligned}$$

Nondeterministically select $r'_1, \dots, r'_n \in T(\mathcal{E}_i(c) \cup \{s_i(c) \simeq t_i(c)\})$ and verify the condition $\mathcal{E}(c), \mathcal{E}(r') \vdash s(r') \simeq t(r')$ (this can be checked in polynomial time using Lemma 2.1).

Such θ is a solution to this system of rigid equations if and only if there is a constant $d \in \mathcal{C}$ such that the following system of rigid w-equations is solvable:

$$\begin{aligned} \mathcal{E}_1(c) &\vdash_{\forall} r'_1 \simeq dx, \\ &\dots \\ \mathcal{E}_n(c) &\vdash_{\forall} r'_n \simeq dx. \end{aligned}$$

Nondeterministically select such d . By Lemma 3.3 we can equivalently replace this system with a system

$$\begin{aligned} \mathcal{E}'_1 &\vdash_{\forall} c_1 \simeq d_1x, \\ &\dots \\ \mathcal{E}'_n &\vdash_{\forall} c_n \simeq d_nx, \end{aligned}$$

where \mathcal{E}'_i are in automaton form. By Lemma 3.4, this system is solvable if and only if the intersection of automata $\mathcal{A}(\mathcal{E}'_1, d_1, c_1), \dots, \mathcal{A}(\mathcal{E}'_n, d_n, c_n)$ is nonempty.

We have given a nondeterministic algorithm reducing monadic SREU with one variable to the DFA intersection nonemptiness problem. On each branch, the algorithm makes polynomially many steps. Applying Lemma 2.4 on the complexity of the DFA intersection nonemptiness problem we get that monadic SREU with one variable is in NPSpace, and hence in PSPACE. \square

Combining Theorem 1 and Lemma 4.3, we obtain

Theorem 3. *Monadic SREU with one variable is PSPACE-complete.*

5. General case

Denote by \mathbf{W} the set of pairs of words on \mathcal{F} . Introduce on \mathbf{W} a binary function $*$, a unary function $'$ and a binary relation \leq in the following way:

$$(U_1, U_2) * (V_1, V_2) \Rightarrow \begin{cases} (U_1, V_2) & \text{if } U_2 = V_1, \\ (V_1, V_2) & \text{otherwise,} \end{cases}$$

$$(U_1, U_2)^r \Rightarrow (U_2, U_1),$$

$$(U_1, U_2) \leq (V_1, V_2) \Rightarrow \text{there is a word } W \text{ such that } (V_1, V_2) = (U_1 W, U_2 W).$$

An *ideal* on \mathbf{W} is any set of pairs containing $(\varepsilon, \varepsilon)$, closed under functions $*$ and r , and upward closed under \leq . The *ideal generated by a set of pairs* S , denoted $\text{ideal}(S)$ is defined as the least ideal containing S .

An *ideal membership question* is an expression

$$(U, V) \in \text{ideal}(\{(U_1, V_1), \dots, (U_n, V_n)\}),$$

where $n \geq 0$ and $U, V, U_1, \dots, U_n, V_1, \dots, V_n \in (\mathcal{F} \cup \mathcal{V})^*$. A *solution to such an ideal membership question* is any word substitution σ such that

1. σ is defined on all variables $U, V, U_1, \dots, U_n, V_1, \dots, V_n$;
2. the word $(U\sigma, V\sigma)$ belongs to the ideal generated by

$$\{(U_1\sigma, V_1\sigma), \dots, (U_n\sigma, V_n\sigma)\}.$$

A *system of ideal membership question* is any finite set of ideal membership questions. A *solution to a system of ideal membership questions* is any substitution that solves each question in the system. The *ideal membership problem* is the decision problem of solvability of systems of ideal membership questions.

The aim of this section is to show that monadic SREU is polynomial-time equivalent to the ideal membership problem. First, we shall one lemma showing that ideal membership questions are at least as expressive as word equations.

Lemma 5.1. *Let U, V be words on $\mathcal{F} \cup \mathcal{V}$. Then the word equation $U \simeq V$ has a solution if and only if the ideal membership question $(U, V) \in \text{ideal}(\emptyset)$ has a solution.*

Proof. Note that $\text{ideal}(\emptyset)$ is the set of all pairs (W, W) . \square

Besides word equations, some other interesting relations on words are also expressible by ideal membership questions. For example, for every words U, V on \mathcal{F} we have $(U, \varepsilon) \in \text{ideal}(\{(V, \varepsilon)\})$ if and only if $U = V^n$ for some natural number n .

The following lemma is the main reason for introducing the notion of an ideal.

Lemma 5.2. *Let $U_1, \dots, U_n, V_1, \dots, V_n, U, V$ be words on \mathcal{F} and a be any constant. Then $aU_1 \simeq aV_1, \dots, aU_n \simeq aV_n \vdash aU \simeq aV$ if and only if $(U, V) \in \text{ideal}(\{(U_1, V_1), \dots, (U_n, V_n)\})$.*

Proof. It is well-known that the set of all logical consequences of a set of ground equations $\{s_1 \simeq t_1, \dots, s_n \simeq t_n\}$ can be characterized as the smallest set of equations \mathcal{E}

such that

1. \mathcal{E} contains all equations of the form $t \simeq t$;
2. $\{s_1 \simeq t_1, \dots, s_n \simeq t_n\} \subseteq \mathcal{E}$;
3. if $(s \simeq t) \in \mathcal{E}$, then $(t \simeq s) \in \mathcal{E}$;
4. if $(r \simeq s) \in \mathcal{E}$ and $(s \simeq t) \in \mathcal{E}$, then $(r \simeq t) \in \mathcal{E}$.
5. if $(s \simeq t) \in \mathcal{E}$ then $(r[s] \simeq r[t]) \in \mathcal{E}$.

When all terms s_i and t_i have the form aW , this characterization immediately implies the statement. \square

Theorem 4. *Monadic SREU is polynomial-time equivalent to the ideal membership problem.*

Proof. By Lemma 2.18, instead of monadic SREU we can consider the problem of solvability of rigid w-equations with one constant. Then use Lemma 5.2. \square

This theorem implies the following.

Theorem 5. *The ideal membership problem is decidable if and only if any of the Problems 1–4 is decidable in the case of monadic signatures.*

5.1. More about the ideal membership problem

In this section we consider the ideal membership problem in more detail. We show that the (un)decidability of this problem is preserved if we add regular constraints (every word variable v_i ranges over a regular set R_i) and the inequality relation.

The proofs in this section will be presented less formally than in the previous sections. First, we show that one can represent regular constraints.

Lemma 5.3. *For every regular set R of words and a variable x there exists a system \mathcal{J} of ideal membership questions such that (i) for every solution θ of \mathcal{J} we have $x\theta \in R$; and (ii) for every $W \in R$ there exists a solution θ of \mathcal{J} such that $x\theta = W$.*

Proof. Similar to Lemma 3.1, for every DFA representing the regular set R , we can effectively find a system of rigid w-equations \mathcal{S} satisfying this condition on solutions. Then use Lemma 5.2 to replace rigid w-equations by ideal membership questions. \square

Second, one can note that inequalities can be added.

Lemma 5.4. *The ideal membership problem is decidable if and only if the ideal membership problem augmented with regular constraints and the inequality constraints $U \neq V$ is decidable.*

Proof. It is enough to note that for each inequality constraint $U \neq V$ we can effectively find a finite set S of systems of rigid w-equations such that for every substitution σ , we

have $U\sigma \neq V\sigma$ if and only if σ is a solution to some member of \mathbf{S} , then by Lemma 5.2 we can replace rigid w-equations by ideal membership questions.

Let $\mathcal{F} = \{f_1, \dots, f_k\}$. We define the following systems of rigid w-equations:

1. Systems \mathcal{S}_i , where $1 \leq i \leq k$:

$$\vdash_{\forall} aU \simeq aVf_iu,$$

where u is a new word variable.

2. Systems \mathcal{S}^i , where $1 \leq i \leq k$:

$$\vdash_{\forall} aUf_iv \simeq aV,$$

where v is a new word variable.

3. Systems \mathcal{S}_{ij}^i , where $1 \leq i, j \leq k$ and $i \neq j$:

$$\vdash_{\forall} aU \simeq awf_iu,$$

$$\vdash_{\forall} aV \simeq awf_jv,$$

where u, v, w are new word variables.

It is easy to see that the set

$$\mathbf{S} = \{\mathcal{S}_i \mid 1 \leq i \leq k\} \cup \{\mathcal{S}^i \mid 1 \leq i \leq k\} \cup \{\mathcal{S}_{ij}^i \mid 1 \leq i, j \leq k \text{ and } i \neq j\}$$

satisfies the statement. \square

Note that the solutions to inequality constraints are searched among words on the original alphabet \mathcal{F} .

Acknowledgements

Remarks of Margus Veanes allowed us to simplify several proofs, and especially that of Theorem 4. We also thank Anatoli Degtyarev and Gennadi Makanin.

References

- [1] A. Degtyarev, Yu. Gurevich, P. Narendran, M. Veanes, A. Voronkov. The decidability of simultaneous rigid E -unification with one variable. UPMail Technical Report 139, Uppsala University, Computing Science Department, March 1997: RTA'98, to appear.
- [2] A. Degtyarev, Yu. Gurevich, A. Voronkov. Herbrand's theorem and equational reasoning: problems and solutions, in: Bulletin of the European Association for Theoretical Computer Science, vol. 60, October 1996, pp. 78–95. The "Logic in Computer Science" column.
- [3] A. Degtyarev, Yu. Matiyasevich, A. Voronkov, Simultaneous rigid E -unification and related algorithmic problems, in: 11th Annu. IEEE Symp. on Logic in Computer Science (LICS'96), New Brunswick, NJ, July 1996, pp. 494–502. IEEE Computer Society Press, Silver Spring, MD.
- [4] A. Degtyarev, A. Voronkov, The undecidability of simultaneous rigid E -unification, Theoret. Comput. Sci. 166(1–2) (1996), 291–300.
- [5] N. Dershowitz, J.-P. Jouannaud, Rewrite systems, in J. Van Leeuwen, (Ed.), Handbook of Theoretical Computer Science, vol. B: Formal Methods and Semantics, chapter 6. North Holland, Amsterdam, 1990. pp. 243–309.

- [6] A. Kościelski, L. Pacholski, Complexity of unification in free groups and free semigroups, in: Proc. 31st Annu. IEEE Symp. on Foundations of Computer Science, Los Alamitos, 1990, pp. 824–829.
- [7] A. Kościelski, L. Pacholski, Complexity of Makanin's algorithm. *J. Assoc. Comput. Mach.* 43(4) (1996), 670–684.
- [8] D. Kozen, Complexity of finitely presented algebras, in: Proc. 9th Annu. Symp. on Theory of Computing, ACM, New York, 1977, pp. 164–177.
- [9] D. Kozen, Lower bounds for natural proof systems, in: Proc. 18th IEEE Symp. on Foundations of Computer Science (FOCS), 1977, pp. 254–266.
- [10] G.S. Makanin, The problem of solvability of equations in free semigroups. *Mat. Sbornik* 103(2) (1977) 147–236. (in Russian). English Translation in *American Mathematical Soc. Translations* (2) (117) (1981).
- [11] D. Perrin, Finite automata, in: J. Van Leeuwen, (Ed.), *Handbook of Theoretical Computer Science*, vol. B: Formal Methods and Semantics, chapter 1. Elsevier, Amsterdam, 1990, pages 1–57.
- [12] D.A. Plaisted, Special cases and substitutes for rigid E -unification, Technical Report MPI-I-95-2-010, Max-Planck-Institut für Informatik, November 1995.
- [13] K.U. Schulz, Makanin's algorithm: two improvements and a generalization, in: K.U. Schulz (Ed.), Proc. 1st Internat. Workshop on Word Equations and Related Topics, Tübingen, number 572 in *Lecture Notes in Computer Science*, 1990.
- [14] R. Shostak, An algorithm for reasoning about equality, *Communications of the ACM*, 21 (1978) 583–585.
- [15] M. Veanes, The undecidability of simultaneous rigid E -unification with two variables, in: G. Gottlob, A. Leitsch, D. Mundici, (Ed.), *Computational Logic and Proof Theory. 5th Kurt Gödel Colloquium, KGC'97*, vol. 1289 of *Lecture Notes in Computer Science*, Vienna, Austria, 1997, pp. 305–318.
- [16] A. Voronkov, Herbrand's theorem, automated reasoning and semantic tableaux, UPMail Technical Report 151, Uppsala University, Computing Science Department, February 1998.
- [17] A. Voronkov, Simultaneous rigid E -unification and other decision problems related to Herbrand's theorem, UPMail Technical Report 152, Uppsala University, Computing Science Department, February 1998.